WHAT IS CLAIMED IS:

1.   A computer program for monitoring the performance of an application by presenting a visual map of the underlying architectural components of the application and the data flow between the architectural components, the computer program comprising software configured to display a visual map having on-screen graphics representing components and data flows of an application, and wherein at least some of the on-screen graphics representing components having similar functionality are organized into groups while other on-screen graphics represent the data flowing between the groups, thereby providing a user with a snapshot overview of the performance of the application.

2.   The computer program of Claim 1, further comprising panels, each panel being associated with one or more of the groups of on-screen graphics representing components having similar functionality, and wherein one or more of the other on-screen graphics represents the data flowing between the panels.

3.   The computer program of Claim 1, wherein the on-screen graphics are grouped within the visual map to resemble the underlying architecture of the application.

4.   The computer program of Claim 1, wherein the application comprises one or more database management systems.

5.   The computer program of Claim 1, wherein the application comprises one or more servers.

6.   The computer program of Claim 1, wherein the application comprises one or more operating systems.

7.   The computer program of Claim 1, wherein the application comprises one or more input/output devices.

QSOFT.004A   Knobbe Martens
                                                                Olson & Bear LLP

8.   The computer program of Claim 1, wherein the application comprises one or more computer accessible storage mediums.

5   9.   The computer program of Claim 1, wherein the application comprises one or more data storage arrays.

10.   The computer program of Claim 1, wherein the application comprises one or more system servers.

10

11.   A method of monitoring a computer program having a plurality of components, the method comprising:

        grouping a plurality of performance data to form an on-screen graphic, wherein the performance data represents the performance of a plurality of

15   program components; and

        changing a graphical attribute of the on-screen graphic when a value of the group of performance data corresponds to a threshold.

12.   The method of Claim 11, wherein the on-screen graphic comprises a

20   hierarchy of graphical caution levels, and wherein the graphical attribute is changed according to the hierarchy.

13.   The method of Claim 11, wherein the on-screen graphic comprises a dataflow.

25

14.   The method of Claim 11, wherein the on-screen graphic comprises alphanumeric text or symbols.

15.   The method of Claim 11, wherein the on-screen graphic comprises a

30   panel.

16.   The method of Claim 11, wherein the on-screen graphic comprises an icon.

-32-

Knobbe Martens
                                                Olson & Bear LLP

17.   The method of Claim 16, wherein the icon comprises a process icon.

18.   The method of Claim 16, wherein the icon comprises a memory icon.

5

19.   The method of Claim 16, wherein the icon comprises a disk icon.

20.   The method of Claim 16, wherein the icon comprises a meter icon.

10   21.   The method of Claim 16, wherein the icon comprises a rotating icon.

22.   The method of Claim 16, wherein the icon comprises a timer icon.

23.   The method of Claim 16, wherein the icon comprises a chart or graph
15   icon.

24.   The method of Claim 11, wherein at least one of the plurality of program
components comprises a process.

20   25.   The method of Claim 11, wherein at least one of the plurality of program
components comprises a flow of data within the program.

26.   The method of Claim 11, wherein at least one of the plurality of program
components comprises a memory structure.

25
27.   The method of Claim 11, wherein at least one of the plurality of program
components comprises a computer accessible storage medium.

28.   The method of Claim 11, wherein the graphical attribute comprises at
30   least one of size, color, texture, text, blinking, existence, speed, acceleration, value,
sound, rotation, animation, and content.

29.   The method of Claim 11, wherein the computer program comprises one
or more database management systems.

-33-

30. The method of Claim 11, wherein the computer program comprises one or more servers.

31. The method of Claim 11, wherein the computer program comprises one or more operating systems.

32. The method of Claim 11, wherein the computer program comprises one or more input/output devices.

33. The method of Claim 11, wherein the computer program comprises one or more computer accessible storage mediums.

34. The method of Claim 11, wherein the computer program comprises one or more data storage arrays.

35. The method of Claim 11, wherein the computer program comprises one or more system servers.

36. A method of monitoring a application program having a plurality of components, the method comprising:

grouping a first plurality of performance data to form a first on-screen graphic, wherein the first performance data represents the performance of a first plurality of program components of an application program;

grouping a second plurality of performance data to form a second on-screen graphic, wherein the second performance data represents the performance of a second plurality of program components of the application program;

grouping the first and second on-screen graphics into a third on-screen graphic; and

changing a graphical attribute of one of the first and second on-screen graphics when a value of the corresponding group of performance data corresponds to a threshold.

-34-

37.   A method of monitoring the performance of a computer program having a plurality of components; the method comprising:

grouping a first plurality of performance data to form a first on-screen graphic, wherein the first performance data represents the performance of a first plurality of program components of a computer program;

grouping a second plurality of performance data to form a second on-screen graphic, wherein the second performance data represents the performance of a second plurality of program components of the computer program;

representing data flowing from the first plurality of program components to the second plurality of program components with an third on-screen graphic; and

changing a graphical attribute of the third on-screen graphic when a value of the data flow corresponds to a threshold.

38.   A method of alerting an user to a potential problem within an application program; the method comprising:

accessing with a monitoring program, performance data representing at least one performance parameter of a monitored program;

comparing the at least one performance parameter to a threshold value associated with the at least one performance parameter; and

accessing a hierarchical set of severity levels to alert a user of the monitoring program, wherein the user is alerted with a first severity level when a threshold value is met and a lesser severity level when the threshold value is not met.

39.   The method of Claim 38, wherein each of the hierarchical set of severity levels defines an action to be performed by the monitoring program.

40.   The method of Claim 39, wherein the action comprises a message to the user.

-35-

Knobbe Martens
                                                        Olson & Bear LLP

41.   The method of Claim 39, wherein the action comprises a change in an attribute of an on-screen graphic of the monitoring program.

5

42.   The method of Claim 39, wherein the action comprises a presentation of additional information corresponding to the monitored program.

43.   The method of Claim 38, wherein the hierarchical set of severity levels includes a middle severity level between the lesser severity level and the first severity

10   level.

44.   The method of Claim 38, wherein meeting the threshold value includes the performance parameter being equal to or greater than the threshold value.

15   45.   The method of Claim 38, wherein the user selects the threshold value.

46.   The method of Claim 38, wherein the user selects one of the hierarchical set of severity levels as the first severity level, and selects another of the hierarchical set of severity levels as the lesser severity level.

20

47.   A monitoring system for monitoring a computer program, the monitoring system comprising:

        a display having a window;

        an on-screen graphic, displayed in the window and representing a

25   plurality of performance statistics corresponding to a plurality of components of a computer program; and

        a severity protocol, associated with the on-screen graphic and configured to set a graphical attribute of the on-screen graphic, wherein the graphical attribute efficiently communicates to a user, the potential or actual existence of

30   performance inhibitors associated with the computer program.

48.   The monitoring system of Claim 47, wherein the computer program comprises one or more database management systems.

-36-
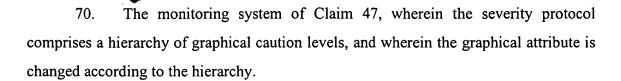
49.   The monitoring system of Claim 47, wherein the computer program comprises one or more servers.

50.   The monitoring system of Claim 47, wherein the computer program comprises one or more operating systems.

51.   The monitoring system of Claim 47, wherein the computer program comprises one or more input/output devices.

52.   The monitoring system of Claim 47, wherein the computer program comprises one or more computer accessible storage mediums.

53.   The monitoring system of Claim 47, wherein the computer program comprises one or more data storage arrays.

54.   The monitoring system of Claim 47, wherein the computer program comprises one or more system servers.

55.   The monitoring system of Claim 47, wherein the on-screen graphic comprises alphanumeric text or symbols.

56.   The monitoring system of Claim 47, wherein the on-screen graphic comprises a dataflow.

57.   The monitoring system of Claim 47, wherein the on-screen graphic comprises a panel.

58.   The monitoring system of Claim 47, wherein the on-screen graphic comprises an icon.

59.   The monitoring system of Claim 58, wherein the icon comprises a process icon.

-37-

Knobbe Martens
                                                                          Olson & Bear LLP

60. The monitoring system of Claim 58, wherein the icon comprises a memory icon.

61. The monitoring system of Claim 58, wherein the icon comprises a disk icon.

62. The monitoring system of Claim 58, wherein the icon comprises a meter icon.

63. The monitoring system of Claim 58, wherein the icon comprises a rotating icon.

64. The monitoring system of Claim 58, wherein the icon comprises a timer icon.

65. The monitoring system of Claim 58, wherein the icon comprises a chart or graph icon.

66. The monitoring system of Claim 47, wherein at least one of the plurality of components of the computer program comprises a process.

67. The monitoring system of Claim 47, wherein at least one of the plurality of components of the computer program comprises a flow of data within the computer program.

68. The monitoring system of Claim 47, wherein at least one of the plurality of components of the computer program comprises a memory structure.

69. The monitoring system of Claim 47, wherein at least one of the plurality of components of the computer program comprises a computer accessible storage medium.

-38-

70.   The monitoring system of Claim 47, wherein the severity protocol comprises a hierarchy of graphical caution levels, and wherein the graphical attribute is changed according to the hierarchy.

5   71.   The monitoring system of Claim 47, wherein the graphical attribute further comprises at least one of size, color, texture, text, blinking, existence, speed, acceleration, value, sound, rotation, animation, and content.

72.   The monitoring system of Claim 47, wherein the performance inhibitors

10   comprise at least one of dataflow bottlenecks, and improperly or inefficiently configured devices, protocols, variables, software modules, or flags.